UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER OF PATENTS AND TRADEMARKS
Washington, D.C. 20231
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/489,570	01/21/2000	William J. Baer	STL000021US1	5998

27896 7590 01/27/2003

EPSTEIN, EDELL, SHAPIRO, FINNAN & LYTLE, LLC
1901 RESEARCH BOULEVARD
SUITE 400
ROCKVILLE, MD 20850

EXAMINER	
PHAM, HUNG Q	
ART UNIT	PAPER NUMBER
2172	

DATE MAILED: 01/27/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/489,570	BAER ET AL.	
	Examiner	Art Unit	
	HUNG Q PHAM	2172	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *19 December 2002* .

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-75* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-75* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

11)☐ The proposed drawing correction filed on _____ is: a)☐ approved b)☐ disapproved by the Examiner.

If approved, corrected drawings are required in reply to this Office action.

12)☐ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

13)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____ .

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).

a) ☐ The translation of the foreign language provisional application has been received.

15)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____ .
4) ☐ Interview Summary (PTO-413) Paper No(s). _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: .

DETAILED ACTION

Continued Prosecution Application

1. The request filed on 12/19/2002 for a Request for Continued Examination (RCE) under 37 CFR 1.114 based on parent Application No. 09/489,570 is acceptable and a RCE has been established.

Applicants canceled claims 76-87. The pending claims are 1-75. Applicants' arguments regarding to the amendment have been fully considered, and an action on the RCE follows.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1-75 are rejected under 35 U.S.C. 103(a) as being unpatentable over DeRose et al. [USP 5,572,722].

Regarding to claims 1, 26, and 51, DeRose teaches a method, a program of instruction and a system for indexing and rendering electronic documents, especially electronic books, having descriptive markup and hierarchical content (Col. 1, lines 10-20). Electronic documents include, but are not limited to, electronic books and operation manuals for large systems, such as for airplane maintenance, etc. The descriptive markup of an input document is interpretable as an ordered hierarchy of content objects. As shown in FIG. 3, a book as ***a content object*** has a plurality of elements, which may have a parent element, a first child element, a last child element, a left sibling element, and a right sibling element (Col. 7, line 59-Col. 8, line 25). As shown in FIG. 6, the data structure element directory 91 is an array of element descriptors 90 is used to improve navigation of a document. Each element descriptor 90 as ***a content entity*** represents an element of the document as ***the content object***. The element descriptor 90 includes a field 92 for representing the parent of the element, a field 94 for representing the first child, a field 96 for representing the last child, a field 98 for representing a left sibling, a field 100 for representing a right sibling, a field 102 for representing the type of the element, and a field 104 for representing the location of text characters for a text chunk or the location of other data associated with the element such as attributes. Those fields that represent elements, such as parent, child and sibling elements, preferably contain the element identifiers assigned to those elements (Col. 9, line 18-Col. 10, line 6). DeRose further discloses that the element directory 91 is created as ***a file object*** by an indexing process in the mass storage device 34 (Col. 10, lines 39-56). The element directory 91 and the process of creating the data structure indicates the

step of ***storing as a file object within the data repository a list of content entity identifiers indicating the content entities within the content object***. DeRose does not explicitly teach: ***the list*** of content entity identifiers ***is manipulable by a user to alter content of the content object;*** and ***storing ones of the plurality of content entities within the data repository as a plurality of individually accessible file objects, wherein each file object contains one content entity***. However, as shown in FIG. 6, field 104 represents the location of text characters for a text chunk or the location of other data associated with the element such as attributes by using a pointer (Col. 9, lines 20-37). As shown in FIG. 8, the process of text chunk begins at step 141 to determine whether the next token is a text chunk. A new element descriptor 90 for the text chunk in the element directory 91 is created at step 142 and the type name for the text chunk is also stored, type name may be reserved name, such as "#TEXT". The text of the text chunk as ***content entity*** is then saved in the open text file in the mass storage device 34, and its location in the text file is recorded in location field 104 of the element descriptor 90 for this text chunk at step 146, the variable EID is incremented in step 148 (Col. 10, lines 39-56 and Col. 12, lines 10-46). Thus, with a unique type names of each text chunk in a document, each text chunk for a unique element descriptor 90 and each pointer for referencing a particular text chunk, a text chunk is written into the open text file in the storage device as a text file object, and the process of creating the text chunks with the texts of each text chunk as content entity indicates the step of ***storing ones of the plurality of content entities within the data repository as a plurality of file objects, wherein each file object contains one content entity*** and by using pointer for referencing a text chunk, a file object of text chunk is

individually accessible. DeRose further discloses: a document having descriptive markup of FIG. 4 may be parsed, and an element directory 91 as in FIG. 6, be generated. This element directory may then be used to traverse the document, since, for each element, the parent element, sibling elements, child elements, and previous elements may be readily accessed in constant time. Such navigation is helpful for combining rendering of the document, full text indexing, generating a table of contents, and creating annotations, bookmarks and history logs. Moreover, since an entry in the element directory may be retrieved in constant time, the element directory and fully-qualified name table may be stored and accessed efficiently on a random-access medium 34 such as a disk (Col. 12, lines 51-65). As shown in FIG. 10 is the method for retrieving and adding a selected word that uses the element directory 91 and the frequency record 152. In step 172, if it was determined that the entry for the variable "P" was not found in the frequency record, the appropriate entry 155 for the element identified by "P" is inserted into the frequency record 152. This insertion involves insuring that the frequency record is an ordered list sorted by element identifier. These relationships hold except for a system, which allows an electronic document to be edited and thus allows changes to the element directory to be made (Col. 13, line 55-Col. 14, line 26). A user may also be provided with the capability of making private and public annotations, bookmarks, history logs and directed paths, which are graphically illustrated in FIG. 22 as part of the document structure. Using such structures a reader of an electronic document may attach comments or other information to a document, create paths through a document. Such capability normally needs to be provided without modifying

the documents being read, because a document may be on a read-only medium, or because a user may not have authority to modify the document, or because different users may be applying modifications to unconnected copies of the document and may wish to share such modifications (Col. 23, lines 54-67). Thus, by using the element directory 91 as an identifier file object containing a list of content entity identifiers, an authorized user could access an element of the document for modifying. This indicates ***the list*** of content entity identifiers ***is manipulable by a user to alter content of the content object***. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the DeRose process to have the steps of storing content entities as a plurality of individually accessible file objects, and by having the content entities as a plurality of individually accessible file objects an electric document such as electric book could be navigated and indexed in accordance with its contents.

Regarding to claims 2, 27 and 52, DeRose teaches all the claimed subject matters as discussed in claims 1, 26 and 51, DeRose further discloses the step of ***creating an attribute table in the data repository for storing an attribute pertaining to at least one of content objects and content entities*** (Fig. 6, Col. 9, lines 21-23 and Col. 10, lines 39-56).

Regarding to claims 3, 28 and 53, DeRose teaches all the claimed subject matters as discussed in claims 2, 27 and 52, DeRose further discloses ***the step of***

creating a row for each content object in the attribute table, the row containing at least one attribute pertaining to the content entity (Fig. 6, Col. 9, line 21-Col. 10. line 6).

Regarding to claims 4, 29 and 54, DeRose teaches all the claimed subject matters as discussed in claims 2, 27 and 52, DeRose further discloses the step of ***creating a row for each content entity in the attribute table, the row containing at least one attribute pertaining to the content object*** (Fig. 6, Col. 9, line 21-Col. 10, line 6).

Regarding to claims 5, 30 and 55, DeRose teaches all the claimed subject matters as discussed in claims 2, 27 and 52, DeRose further discloses: ***at least one attribute is extracted from the content object*** (Col. 9, lines 21-37).

Regarding to claims 6, 31 and 56, DeRose teaches all the claimed subject matters as discussed in claims 1, 26 and 51, but fails to disclose: ***ones of the content entities further comprise components associated with the content object, and further comprising the step of storing each associated component as a file object***. However, as shown in Fig. 3, the body 50 of book 52 comprises an artwork 64 that associated with book 52 as the content object. DeRose further discloses: some of the tags in the descriptive markup of the document may also be empty tags such as tag 49 (FIG. 4). Such empty tags may be used for cross-referencing, referencing other documents, or for referencing graphic or other types of non-text information, etc. These tags often have attributes which are variables, such as "file", to which are assigned values, such as

"myfig12" (Col. 8, lines 61-67). This indicates ones of the content entities further comprise components associated with the content object, and further comprising the step of storing each associated component as a file object. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the DeRose method to include the step of storing each associated component as a file object in order to format an electric document such as electric book in accordance with its non-text information such as image file.

Regarding to claims 7, 32 and 57, DeRose teaches all the claimed subject matters as discussed in claims 1, 26 and 51, DeRose further discloses: ***the content object is one of a book, a collection of images, an album, and a video*** (Col. 7, lines 59-64).

Regarding to claims 8, 33 and 58, DeRose teaches all the claimed subject matters as discussed in claims 1, 26 and 51, DeRose further discloses: ***the content object is a book and ones of the content entities are one of volumes, chapters and sections*** (Col. 7, lines 59-64).

Regarding to claims 9, 34 and 59, DeRose teaches all the claimed subject matters as discussed in claims 1, 26 and 51, but fails to disclose: ***the content object is a compilation of content***. However, as shown in Fig. 3 and 5, the book is a content object comprises title, body, chapters and sections. This indicates the content object is a compilation of content. Therefore, it would have been obvious for one of ordinary skill in

the art at the time the invention was made to modify the DeRose method to include the step of compilation of content into the content object in order to format an electric document such as electric book in accordance with its contents.

Regarding to claims 10, 35 and 60, DeRose teaches all the claimed subject matters as discussed in claims 6, 31 and 56, DeRose further discloses: ***at least one of the associated components comprises an image*** (Col. 8, lines 18-25).

Regarding to claims 11, 36 and 61, DeRose teaches a method, a program of instruction and a system for indexing and rendering electronic documents, especially electronic books, having descriptive markup and hierarchical content (Col. 1, lines 10-20). Electronic documents include, but are not limited to, electronic books and operation manuals for large systems, such as for airplane maintenance, etc. The descriptive markup of an input document is interpretable as an ordered hierarchy of content objects. As shown in FIG. 3, a book as ***a content object*** has a plurality of elements, which may have a parent element, a first child element, a last child element, a left sibling element, and a right sibling element (Col. 7, line 59-Col. 8, line 25). As shown in FIG. 6, the data structure element directory 91 is an array of element descriptors 90 is used to improve navigation of a document. Each element descriptor 90 as ***a content entity*** represents an element of the document as ***the content object***. The element descriptor 90 includes a field 92 for representing the parent of the element, a field 94 for representing the first child, a field 96 for representing the last child, a field 98 for representing a left

sibling, a field 100 for representing a right sibling, a field 102 for representing the type of the element, and a field 104 for representing the location of text characters for a text chunk or the location of other data associated with the element such as attributes. Those fields that represent elements, such as parent, child and sibling elements, preferably contain the element identifiers assigned to those elements (Col. 9, line 18-Col. 10, line 6). DeRose further discloses that the element directory 91 is created as ***a file object*** by an indexing process in the mass storage device 34 (Col. 10, lines 39-56). The element directory 91 and the process of creating the data structure indicates the step of ***storing as a file object within the data repository an outline of containers and content entity identifiers defining the content and hierarchy of the content object***. DeRose does not explicitly teach the step of ***storing ones of the plurality of content entities within the data repository as a plurality of individually accessible file objects, wherein each file object contains one content entity***. However, as shown in FIG. 6, field 104 represents the location of text characters for a text chunk or the location of other data associated with the element such as attributes by using a pointer (Col. 9, lines 20-37). As shown in FIG. 8, text content is created as a file object and the process of text chunk begins at step 141 to determine whether the next token is a text chunk. A new element descriptor 90 for the text chunk in the element directory 91 is created at step 142 in a manner similar to step 116 and the type name for the text chunk is also stored, type name may be reserved name, such as "#TEXT". The text of the text chunk as ***content entity*** is then saved in the open text file in the mass storage device 34, and its location in the text file is recorded in location field 104 of the element descriptor 90 for this text chunk at step

146, the variable EID is incremented in step 148 (Col. 10, lines 39-56 and Col. 12, lines 10-46). Thus, with a unique type names of each text chunk in a document, each text chunk for a unique element descriptor 90 and each pointer for referencing a particular text chunk, a text chunk is written into the open text file in the storage device as a text file object, and the process of creating the text chunks with the texts of each text chunk as content entity indicates the step of ***storing ones of the plurality of content entities within the data repository as a plurality of file objects, wherein each file object contains one content entity*** and by using pointer for referencing a text chunk, a file object of text chunk is ***individually accessible***. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the DeRose process to have a plurality of content file objects, and list of content entity is manipulable for altering the content of the content object, and by having a plurality of content file objects, and list of content entity, an electric document such as electric book could be navigated, indexed and modified in accordance with its contents.

Regarding to claims 12, 37 and 62, DeRose teaches all the claimed subject matters as discussed in claims 11, 36 and 61, DeRose further discloses the step of ***creating an attribute table in the data repository for storing an attribute pertaining to at least one of content objects and content entities*** (Fig. 6, Col. 9, lines 21-23 and Col. 10, lines 39-56).

Regarding to claims 13, 38 and 63, DeRose teaches all the claimed subject matters as discussed in claims 12, 37 and 62, DeRose further discloses the step of ***creating a row for each content object in the attribute table, the row containing at least one attribute pertaining to the content object*** (Fig. 6, Col. 9, line 21-Col. 10. line 6).

Regarding to claims 14, 39 and 64, DeRose teaches all the claimed subject matters as discussed in claims 12, 37 and 62, DeRose further discloses the step of ***creating a row for each container in the attribute table, the row containing at least one attribute pertaining to the container*** (Fig. 6, Col. 9, line 21-Col. 10, line 6).

Regarding to claims 15, 40 and 65, DeRose teaches all the claimed subject matters as discussed in claims 12, 37 and 62, DeRose further discloses the step of ***creating a row for each content entity in the attribute table, the row containing at least one attribute pertaining to the content entity*** (Fig. 6, Col. 9, line 21-Col. 10, line 6).

Regarding to claim 16, 41 and 66, DeRose teaches all the claimed subject matters as discussed in claims 12, 37 and 62, DeRose further discloses: ***at least one attribute is extracted from the content object*** (Col. 9, lines 21-37).

Regarding to claims 17, 42 and 67, DeRose teaches all the claimed subject matters as discussed in claims 11, 36 and 61, but fails to disclose: ***ones of the content entities further comprise components associated with the content object, and further***

comprising the step of storing each associated component as a file object. However, as shown in Fig. 3, the body 50 of book 52 comprises an artwork 64 that associated with book 52 as the content object. DeRose further discloses: some of the tags in the descriptive markup of the document may also be empty tags such as tag 49 (FIG. 4). Such empty tags may be used for cross-referencing, referencing other documents, or for referencing graphic or other types of non-text information, etc. These tags often have attributes which are variables, such as "file", to which are assigned values, such as "myfig12" (Col. 8, lines 61-67). This indicates ones of the content entities further comprise components associated with the content object, and further comprising the step of storing each associated component as a file object. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the DeRose method to include the step of storing each associated component as a file object in order to format an electric document such as electric book in accordance with its non-text information such as image file.

Regarding to claims 18, 43 and 68, DeRose teaches all the claimed subject matters as discussed in claims 11, 36 and 61, DeRose further discloses: ***the content object is one of a book, a collection of images, an album, and a video*** (Col. 7, lines 59-64).

Regarding to claims 19, 44 and 69, DeRose teaches all the claimed subject matters as discussed in claims 11, 36 and 61, DeRose further discloses: ***the content***

object is a book and ones of the content entities are one of volumes, chapters and sections (Col. 7, lines 59-64).

Regarding to claims 20, 45 and 70, DeRose teaches all the claimed subject matters as discussed in claims 11, 36 and 61, DeRose further discloses: ***the content object is a book and ones of the containers are one of books, volumes and chapters*** (Col. 7, lines 59-64).

Regarding to claims 21, 46 and 71, DeRose teaches all the claimed subject matters as discussed in claims 11, 36 and 61, but fails to disclose: ***the content object is a compilation of content***. However, as shown in Fig. 3 and 5, the book is a content object comprises title, body, chapters and sections. This indicates the content object is a compilation of content. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the DeRose method to include the step of compilation of content into the content object in order to format an electric document such as electric book in accordance with its contents.

Regarding to claims 22, 47 and 72, DeRose teaches all the claimed subject matters as discussed in claims 14, 42 and 64, DeRose further discloses: ***at least one of the associated components comprises one of an image, a video segment and an audio segment*** (Col. 8, lines 18-25).

Regarding to claims 23, 48 and 73, DeRose teaches a method, a program of instruction and a system for indexing and rendering electronic documents, especially electronic books, having descriptive markup and hierarchical content (Col. 1, lines 10-20). Electronic documents include, but are not limited to, electronic books and operation manuals for large systems, such as for airplane maintenance, etc. The descriptive markup of an input document is interpretable as an ordered hierarchy of content objects. As shown in FIG. 3, a book as ***a content object*** has a plurality of elements, which may have a parent element, a first child element, a last child element, a left sibling element, and a right sibling element (Col. 7, line 59-Col. 8, line 25). As shown in FIG. 6, the data structure element directory 91 is an array of element descriptors 90 is used to improve navigation of a document. Each element descriptor 90 as ***a content entity*** represents an element of the document as ***the content object***. The element descriptor 90 includes a field 92 for representing the parent of the element, a field 94 for representing the first child, a field 96 for representing the last child, a field 98 for representing a left sibling, a field 100 for representing a right sibling, a field 102 for representing the type of the element, and a field 104 for representing the location of text characters for a text chunk or the location of other data associated with the element such as attributes. Those fields that represent elements, such as parent, child and sibling elements, preferably contain the element identifiers assigned to those elements (Col. 9, line 18-Col. 10, line 6). DeRose further discloses that the element directory 91 and the text content as content entity is created as ***a file object*** by an indexing process in the mass storage device 34 (Col. 10, lines 39-56) in order to generate a table of contents, create

annotation, bookmarks (Col. 12, lines 51-61). As shown in FIG. 12-14 is the process of retrieving an electronic book after the element directory 91 being generated as the step of ***retrieving the file object containing the list of content entity identifiers within the data repository***. DeRose teaches that rendering includes processes for selecting a point in the document from which rendering may begin, displaying the document on an output device, and other operations to be performed by a reader of the electronic document. By using a table of contents from which a section of a document may be selected. A user may also have a directed path, bookmark, history log or other list of pre-selected starting points (Col. 15, lines 37-56). When the table of contents is displayed on the screen, as in FIGS. 12-14, the title for the first element in the table of contents file is displayed. A section of the table of contents may then be expanded, for example, responsive to a mouse event or other indication by a user, by displaying the titles for any immediate sub-elements of a selected displayed element and for subsequent elements, which were in the original display (Col. 16, line 59-Col. 17, line 26). The rendering process as discussed above indicates the step of ***for each content entity identifier, retrieving the file object corresponding to the identified content entity; and inserting the content entity into the ordered list at the location of its content entity identifier***. DeRose does not explicitly teach: ***each content entity is stored as an individually accessible file object*** and the file object is retrieved is ***the individually accessible file object***. However, as shown in FIG. 6, field 104 represents the location of text characters for a text chunk or the location of other data associated with the element such as attributes by using a pointer (Col. 9, lines 20-37). As shown in FIG. 8, the process of text chunk begins at step 141 to

determine whether the next token is a text chunk. A new element descriptor 90 for the text chunk in the element directory 91 is created at step 142 and the type name for the text chunk is also stored, type name may be reserved name, such as "#TEXT". The text of the text chunk as ***content entity*** is then saved in the open text file in the mass storage device 34, and its location in the text file is recorded in location field 104 of the element descriptor 90 for this text chunk at step 146, the variable EID is incremented in step 148 (Col. 10, lines 39-56 and Col. 12, lines 10-46). Thus, with a unique type names of each text chunk in a document, each text chunk for a unique element descriptor 90 and each pointer for referencing a particular text chunk, a text chunk is written into the open text file in the storage device as a text file object, and the process of creating the text chunks with the texts of each text chunk as content entity, a file object of text chunk is stored and retrieved as an ***individually accessible file object***. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the DeRose process to have the steps of storing content entity as an individually accessible file objects, and by having the content entity as an individually accessible file objects an electric document such as electric book could be navigated and indexed in accordance with its contents.

Regarding to claims 24, 49 and 74, DeRose teaches a method, a program of instruction and a system for indexing and rendering electronic documents, especially electronic books, having descriptive markup and hierarchical content (Col. 1, lines 10-20). Electronic documents include, but are not limited to, electronic books and operation

manuals for large systems, such as for airplane maintenance, etc. The descriptive markup of an input document is interpretable as an ordered hierarchy of content objects. As shown in FIG. 3, a book as ***a content object*** has a plurality of elements, which may have a parent element, a first child element, a last child element, a left sibling element, and a right sibling element (Col. 7, line 59-Col. 8, line 25). As shown in FIG. 6, the data structure element directory 91 is an array of element descriptors 90 is used to improve navigation of a document. Each element descriptor 90 as ***a content entity*** represents an element of the document as ***the content object***. The element descriptor 90 includes a field 92 for representing the parent of the element, a field 94 for representing the first child, a field 96 for representing the last child, a field 98 for representing a left sibling, a field 100 for representing a right sibling, a field 102 for representing the type of the element, and a field 104 for representing the location of text characters for a text chunk or the location of other data associated with the element such as attributes. Those fields that represent elements, such as parent, child and sibling elements, preferably contain the element identifiers assigned to those elements (Col. 9, line 18-Col. 10, line 6). DeRose further discloses that the element directory 91 and the text content as content entity is created as ***a file object*** by an indexing process in the mass storage device 34 (Col. 10, lines 39-56) in order to generate a table of contents, create annotation, bookmarks (Col. 12, lines 51-61). As shown in FIG. 12-14 is the process of retrieving an electronic book after the element directory 91 being generated. DeRose teaches that rendering includes processes for selecting a point in the document from which rendering may begin, displaying the document on an output device, and other

operations to be performed by a reader of the electronic document. By using a table of contents from which a section of a document may be selected. A user may also have a directed path, bookmark, history log or other list of pre-selected starting points (Col. 15, lines 37-56). When the table of contents is displayed on the screen as ***the contents of the content object being defined by an ordered list of content entity identifier identifying one or more content entities***, as in FIGS. 12-14, the title for the first element in the table of contents file is displayed. A section of the table of contents may then be expanded, for example, responsive to a mouse event or other indication by a user, by displaying the titles for any immediate sub-elements of a selected displayed element and for subsequent elements, which were in the original display (Col. 16, line 59-Col. 17, line 26). The rendering process as discussed above indicates the step of ***for each content entity identifier, retrieving the file object corresponding to the identified content entity; and inserting the content entity into the ordered list at the location of its content entity identifier***. DeRose does not explicitly teach: each content entity is ***an individually accessible file object***. However, as shown in FIG. 6, field 104 represents the location of text characters for a text chunk or the location of other data associated with the element such as attributes by using a pointer (Col. 9, lines 20-37). As shown in FIG. 8, the process of text chunk begins at step 141 to determine whether the next token is a text chunk. A new element descriptor 90 for the text chunk in the element directory 91 is created at step 142 and the type name for the text chunk is also stored, type name may be reserved name, such as "#TEXT". The text of the text chunk as ***content entity*** is then saved in the open text file in the mass storage device 34, and its location in the text file

is recorded in location field 104 of the element descriptor 90 for this text chunk at step 146, the variable EID is incremented in step 148 (Col. 10, lines 39-56 and Col. 12, lines 10-46). Thus, with a unique type names of each text chunk in a document, each text chunk for a unique element descriptor 90 and each pointer for referencing a particular text chunk, a text chunk is written into the open text file in the storage device as a text file object, and the process of creating the text chunks with the texts of each text chunk as content entity, a file object of text chunk is stored as an ***individually accessible file object***. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the DeRose process to have the steps of storing content entity as an individually accessible file objects, and by having the content entity as an individually accessible file objects an electric document such as electric book could be navigated and indexed in accordance with its contents.

Regarding to claims 25, 50 and 75, DeRose teaches all the claimed subject matters as discussed in claims 24, 49 and 74, DeRose further discloses the step of ***assigning an identifier to the content object; and assigning new content entity identifiers to the content entities, the new identifiers including the identifier assigned to the content object*** (Fig. 6, Col. 9).

Conclusion

4. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Hung Pham whose telephone number is 703-605 4242. The examiner can normally be reached on Monday-Friday, 7:00 Am - 3:30 Pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, VU, KIM YEN can be reached on 703-305 4393. The fax phone numbers for the organization where this application or proceeding is assigned are 703-746 7239 for regular communications and 703-746 7238 for After Final communications. Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is 703-305 3900.

Examiner: Hung Pham
January 15, 2003

JEAN M. CORRIELUS
PRIMARY EXAMINER